AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method for logging file system operations, comprising:

receiving a request to perform a file system operation at a primary server;

making a call to an underlying file system to perform the file system operation; and

logging the file system operation to a log within a log device to facilitate recovery of the file system operation in the event of a system failure before the file system operation is committed to non-volatile storage, wherein the log device is located on a secondary server that is separate from the primary server, and wherein the secondary server acts as a backup for the primary server;

wherein the file system operation includes arguments and data needed to repeat the file system operation; and

wherein locating the log on the secondary server facilitates failover to the secondary server when the primary server fails; and

wherein locating the log in volatile memory on the secondary server facilitates recovery of the file system operation without adding delay to normal file system operations due to writes to non-volatile storage;

periodically committing the log to the underlying file system in non-volatile storage, and removing outstanding file system operations from the log in volatile memory;

and

wherein upon a subsequent computer system startup, the method further comprises:

examining the log within the log device; and

replaying any file system operations from the log~~,~~

~~wherein the log is only in volatile memory~~ that are in volatile memory and that have not been committed to non-volatile storage.

2. (Original) The method of claim 1, wherein logging the file system operation involves storing an identifier for the file system operation to the log device.

3. (Currently amended) The method of claim 1~~, further comprising~~ wherein periodically committing the log to the underlying file system ~~by~~ further comprises:

freezing ongoing activity on a file system;

making a call to the underlying file system to flush memory buffers to non-volatile storage, whereby outstanding file system operations are guaranteed to be committed to non-volatile storage;

~~removing outstanding file system operations from the log; and~~

unfreezing the ongoing activity on the file system.

4 (Canceled).

5. (Original) The method of claim 1, further comprising checking for dependencies between the file system operation and ongoing file system operations; and

if dependencies are detected, ensuring that the file system operation and the ongoing file system operations complete in an order that satisfies the dependencies.

6 (Canceled).

7. (Original) The method of claim 1, further comprising:

associating the file system operation with a transaction identifier for a set of related file system operations; and

wherein logging the file system operation involves storing the file system operation with the transaction identifier to the log device.

8. (Original) The method of claim 1, wherein logging the file system operation involves:

determining if the file system operation belongs to a subset of file system operations that are subject to logging; and

if so, logging the file system operation.

9. (Original) The method of claim 8, wherein the subset of file system operations are non-idempotent file system operations.

10. (Original) The method of claim 1, wherein the log device stores the file system operation in volatile storage.

11. (Original) The method of claim 1, wherein the log device stores the file system operation in non-volatile storage.

12. (Currently amended) A computer-readable storage medium storing instructions that when executed by a computer cause the computer to perform a method for logging file system operations, wherein the computer-readable storage medium includes one of a volatile memory, a non-volatile memory, a disk drive, a magnetic tape, a compact disc, a digital versatile disc, and a digital video disk, the method comprising:

receiving a request to perform a file system operation at a primary server in a highly available system;

making a call to an underlying file system to perform the file system operation; and

logging the file system operation to a log within a log device to facilitate recovery of the file system operation in the event of a system failure before the file system operation is committed to non-volatile storage, wherein the log device is located on a secondary server that is separate from the primary server in the highly available system and wherein the secondary server acts as a backup for the primary server;

wherein the file system operation includes arguments and data needed to repeat the file system operation; and

wherein locating the log on the secondary server facilitates failover to the secondary server when the primary server fails; and

wherein locating the log in volatile memory on the secondary server facilitates recovery of the file system operation without adding delay to normal file system operations due to writes to non-volatile storage;

periodically committing the log to the underlying file system in non-volatile storage, and removing outstanding file system operations from the log in volatile memory;

and

wherein upon a subsequent computer system startup, the method further comprises:

examining the log within the log device; and

replaying any file system operations from the log~~, wherein the log is only in volatile memory~~ that are in volatile memory and that have not been committed to non-volatile storage.

13. (Original) The computer-readable storage medium of claim 12, wherein logging the file system operation involves storing an identifier for the file system operation to the log device.

14. (Currently amended) The computer-readable storage medium of claim 12, wherein ~~the method further comprises~~ periodically committing the log to the underlying file system ~~by~~ further comprises:

freezing ongoing activity on a file system;

making a call to the underlying file system to flush memory buffers to non-volatile storage, whereby outstanding file system operations are guaranteed to be committed to non-volatile storage;

~~removing outstanding file system operations from the log;~~ and

unfreezing the ongoing activity on the file system.

15 (Canceled).

16. (Original) The computer-readable storage medium of claim 12, wherein the method further comprises checking for dependencies between the file system operation and ongoing file system operations; and

if dependencies are detected, ensuring that the file system operation and the ongoing file system operations complete in an order that satisfies the dependencies.

17 (Canceled).

18. (Original) The computer-readable storage medium of claim 12, wherein the method further comprises:

associating the file system operation with a transaction identifier for a set of related file system operations; and

wherein logging the file system operation involves storing the file system operation with the transaction identifier to the log device.

19. (Original) The computer-readable storage medium of claim 12, wherein logging the file system operation involves:

determining if the file system operation belongs to a subset of file system operations that are subject to logging; and

if so, logging the file system operation.

20. (Original) The computer-readable storage medium of claim 19, wherein the subset of file system operations are non-idempotent file system operations.

21. (Original) The computer-readable storage medium of claim 12, wherein the log device stores the file system operation in volatile storage.

22. (Original) The computer-readable storage medium of claim 12, wherein the log device stores the file system operation in non-volatile storage.

23. (Currently amended) An apparatus that logs file system operations, comprising:

a receiving mechanism that is configured to receive a request to perform a file system operation at a primary server in a highly available system;

a calling mechanism that is configured to make a call to an underlying file system to perform the file system operation; and

a logging mechanism that is configured to log the file system operation to a log within a log device to facilitate recovery of the file system operation in the event of a system failure before the file system operation is committed to non-volatile storage, wherein the log device is located on a secondary server that is separate from the primary server in the highly available system and wherein the secondary server acts as a backup for the primary server;

wherein the file system operation includes arguments and data needed to repeat the file system operation; and

wherein locating the log on the secondary server facilitates failover to the secondary server when the primary server fails; and

wherein locating the log in volatile memory on the secondary server facilitates recovery of the file system operation without adding delay to normal file system operations due to writes to non-volatile storage;

a committing mechanism for periodically committing the log to the underlying file system in non-volatile storage, and removing outstanding file system operations from the log in volatile memory;

and

a recovery mechanism that operates during system startup, wherein the recovery mechanism is configured to:

examine the log within the log device; and to

replay any file system operations from the log~~, wherein the log is only in volatile memory~~ that are in volatile memory and that have not been committed to non-volatile storage.

24. (Original) The apparatus of claim 23, wherein the logging mechanism is configured to store an identifier for the file system operation to the log device.

25. (Currently amended) The apparatus of claim 23, wherein ~~the logging mechanism is configured to periodically~~ the committing mechanism further comprises:

~~freeze~~ freezing ongoing activity on a file system;

~~make~~ making a call to the underlying file system to flush memory buffers to non-volatile storage, whereby outstanding file system operations are guaranteed to be committed to non-volatile storage;

~~remove outstanding file system operations from the log;~~ and ~~to~~ ~~unfreeze~~ unfreezing the ongoing activity on the file system.

26 (Canceled).

27. (Original) The apparatus of claim 23, further comprising a dependency handler that is configured to:

check for dependencies between the file system operation and ongoing file system operations; and to

ensure that the file system operation and the ongoing file system operations complete in an order that satisfies dependencies if dependencies are detected.

28 (Canceled).

29. (Original) The apparatus of claim 23, further comprising a transaction mechanism that is configured to associate the file system operation with a transaction identifier for a set of related file system operations; and

wherein the logging mechanism is configured to log the file system operation with the transaction identifier to the log device.

30. (Original) The apparatus of claim 23, wherein the logging mechanism is configured to:

determine if the file system operation belongs to a subset of file system operations that are subject to logging; and to

log the file system operation if the file system operation belongs to the subset of file system operations that are subject to logging.

31. (Original) The apparatus of claim 30, wherein the subset of file system operations are non-idempotent file system operations.

32. (Original) The apparatus of claim 23, wherein the log device is configured to store the file system operation in volatile storage.

33. (Original) The apparatus of claim 23, wherein the log device is configured to store the file system operation in non-volatile storage.

34. (Previously presented) The method of claim 1, wherein the primary server is in a highly available system; and wherein the secondary server is in the highly available system.

35. (Previously presented) The computer-readable storage medium of claim 12, wherein the primary server is in a highly available system; and wherein the secondary server is in the highly available system.

36. (Previously presented) The apparatus of claim 23, wherein the primary server is in a highly available system; and wherein the secondary server in the highly available system.